

面向复杂路网低频 GPS 采样数据的新型地图匹配算法*

黄振锋^{1a}, 乔少杰^{1b,1c†}, 韩楠^{1d}, 元昌安^{2,3}, 许源平^{1b}, 曹亮^{1b,1c}, 覃晓³, 魏军林^{1b}

(1. 成都信息工程大学 a. 网络空间安全学院; b. 软件工程学院; c. 软件自动生成与智能服务四川省重点实验室; d. 管理学院, 成都 610103; 2. 广西教育学院, 南宁 530007; 3. 南宁师范大学 计算机与信息工程学院, 南宁 530299)

摘要: 大数据时代低频采样交通轨迹数据呈指数级增长, 准确、高效地对复杂道路网络中产生的海量低频浮动车数据进行地图匹配对出租车载客热点和路线推荐具有重要意义。基于上述考虑, 提出了一种基于曲线拟合的改进算法, 对缺失的轨迹数据和道路网络数据分别使用插值和均值化的方法进行补全, 利用 Geohash 技术对道路网络和轨迹数据进行存储和搜索, 充分考虑车辆速度和道路限速因素, 使用轨迹点后向量和路段向量对路候选段进行分析, 设计综合评价函数得到最优匹配结果。实验结果表明, 与传统垂直投影算法和曲线拟合算法进行对比, 所提曲线拟合算法准确率较高, 时间效率得到显著提升。

关键词: 地图匹配; 复杂道路网络; 低频浮动车数据; 曲线拟合; Geohash

中图分类号: TP301.6 **doi:** 10.19734/j.issn.1001-3695.2019.10.0601

Novel map-matching algorithm based on low-frequency GPS sampling data in complex road networks

Huang Zhenfeng^{1a}, Qiao Shaojie^{1b,1c†}, Han Nan^{1d}, Yuan Chang'an^{2,3}, Xu Yuanping^{1b}, Cao Liang^{1b,1c}, Qin Xiao³, Wei Junlin^{1b}

(1. a. School of Cybersecurity, b. School of Software Engineering, c. Software Automatic Generation & Intelligent Service Key Laboratory of Sichuan Province, d. School of Management, Chengdu University of Information Technology, Chengdu 610103, China; 2. Guangxi College of Education, Nanning 530007, China; 3. School of Computer & Information Engineering, Nanning Normal University, Nanning 530299, China)

Abstract: The number of low-frequency sampling traffic trajectory data is growing explosively in big data era. Map matching techniques play an essential role in taxi pick-up hotspot and route recommendation, which can help accurately and efficiently matching low-frequency floating car data from complex road networks. In view of the aforementioned discussion, this paper proposed an improved algorithm based on curve fitting. This algorithm fills missing trajectory data and road network data by interpolation and mean methods, employs Geohash technology to store and search road network data and trajectory data, takes into full consideration the factors of the speed of vehicles and road speed limit, uses the backward vector of trajectory point and road segment vector to analyze the candidate road segments, and designs a comprehensive evaluation function to obtain the optimal matching results. The experimental results showed that the proposed curve-fitting algorithm has a higher matching accuracy and faster time efficiency when compared with the traditional vertical projection and curve-fitting algorithms.

Key words: map-matching; complex road networks; low-frequency floating car data; curve fitting; geohash

0 引言

随着基于位置的服务技术(LBS, location-based service)的兴起和 GPS 数据采集技术广泛使用, 交通轨迹数据呈爆炸式增长。在交通轨迹数据中, 地图匹配技术是浮动车技术中重要的一个部分, 将附带经纬度、时间、角度等一系列信息作为判断依据, 把车辆的 GPS 点匹配到正确的电子地图道路上以便确认车辆正确的行驶轨迹。当前, 由于北斗卫星和车载位置存在定位误差, 车辆 GPS 的位置和电子地图中道路位置存在偏差, 因此需要把车辆定位位置纠正到对应道路上。此外, 基于海量浮动车数据分析挖掘, 能够提取出诸多对交通有利的数据^[1], 如: 出租车载客热点发现、城市道路更新、交通路段拥堵情况预测等。上述应用的前提是 GPS 轨迹点匹配到正确的道路上。

研究动机: 大数据时代, 浮动车数据增长迅速, 将海量的浮动车数据中进行地图匹配操作, 会消耗大量的时间成本和计算资源。考虑通信和能源成本等因素, 大多数出租车公司采用低频 GPS 采样技术^[2]。以基于低频 GPS 采样技术的上海出租车为例, 每辆出租车从当天凌晨 12 点到次日, 采集到的浮动车数据有 1500 多条。根据 2018 年全国出租车数量统计, 上海出租车数量为 48900, 一个城市出租车每天产生的数据信息达到千万条。传统的地图匹配算法大多数使用高频 GPS 采样的数据进行实验, 针对低频采样技术的浮动车数据地图匹配研究较少。此外, 没有综合考虑正确率和时间效率两个因素。针对海量低频浮动车数据, 需要提出一种正确、快速、普适的地图匹配方法。

根据车载数据不同的采集频率, 地图匹配技术可划分为基于高频采样的地图匹配算法和基于低频采样的地图匹配算

收稿日期: 2019-10-24; **修回日期:** 2019-11-28 **基金项目:** 四川省高校人文社会科学重点研究基地科研项目(JGYQ2018010); 国家自然科学基金资助项目(61802035, 61772091, 61962006); 四川省科技计划资助项目(2018JY0448, 2019YFG0106, 2019YFS0067); 四川省重大科技专项项目(2017GZDZX0002, 2018GZDZX0049); 广西自然科学基金资助项目(2018GXNSFDA138005); 四川高校科研创新团队设计计划资助项目(18TD0027)

作者简介: 黄振锋(1996-), 男, 广东阳江人, 硕士研究生, 主要研究方向为数据挖掘; 乔少杰(1981-), 男(通信作者), 山东招远人, 教授, 博士, 主要研究方向为移动数据库, 人工智能(sjqiao@cuit.edu.cn); 韩楠(1984-), 女, 陕西宝鸡人, 副教授, 博士, 主要研究方向为轨迹数据挖掘; 元昌安(1964-), 男, 安徽肥东人, 教授, 博导, 博士, 主要研究方向为人工智能; 许源平, (1980-), 男, 四川成都人, 教授, 博士, 主要研究方向为机器视觉与智能计算; 曹亮(1985-), 男, 江西九江, 讲师, 硕士, 主要研究方向为软件工程、大数据; 覃晓(1973-), 女, 广西环江人, 副教授, 硕士, 主要研究方向为人工智能、图像处理; 魏军林(1963-), 男, 甘肃临洮人, 高级工程师, 硕士, 主要研究方向为 GIS 和时空数据分析。

法, 由于大多数出租车公司采用低频的采样技术, 所以侧重基于低频采样数据的地图匹配算法。传统几何匹配算法中点到点^[3]、点到线^[4]、线到线^[5]匹配方法, 由于仅考虑待匹配点到候选道路的距离因素, 匹配速度快, 无论在高频采样还是低频采样数据中都容易造成匹配错误, 实用性不强。使用拓扑关系^[6,7]的匹配算法中, 由于限制待匹配点的候选道路, 很大程度能提升匹配的正确性, 但对限制候选道路要求较高。使用拓扑关系需要考虑道路数据的拓扑可达性, 一般需要进行局部搜索甚至全局搜索, 时间开销较大。此外, 低频采样的数据由于过于稀疏, 导致拓扑搜索失败, 容易发生匹配错误。使用概率统计的置信区间算法^[8,9]、模糊逻辑算法^[10]、D-S 证据推理^[11]方法等, 上述算法虽然匹配精度高, 但由于算法实现复杂, 计算开销大, 不适合在快速的匹配算法中使用。

为了提高大规模浮动车地图匹配的准确性和效率性, 主要从以下几个方面作出贡献: a) 对浮动车数据进行预处理, 剔除冗余、异常数据, 使用插值补充的方法对稀疏 GPS 数据轨迹进行补充, 将处理后 GPS 轨迹数据坐标系统一转换为与道路数据相同的坐标系; b) 对原始城市道路数据进行预处理, 合并所有不同类型的道路数据, 对缺失字段值的道路进行补充, 对道路进行拓扑构建; c) 利用 Geohash 存储和搜索技术, 对道路网络数据和轨迹数据进行压缩存储。为了增加路段折点搜索的正确性, 通过扩展搜索的办法对附近相邻区域遍历搜索。此外, 使用后向夹角的方法对候选路段进行筛选, 进一步提高算法的运行效率; d) 在曲线拟合算法的基础上, 提出 SP(speed perfection)曲线拟合算法, 充分考虑车辆和道路的速度因素影响, 调整道路夹角、距离、车辆速度的权重提高道路匹配的准确性; e) 利用匹配准确率和运行时间两个指标, 基于 GPS 模拟数据和车辆真实数据, 对 SP 曲线拟合算法和传统的垂直投影算法、曲线拟合算法进行对比实验, 验证提出的算法性能优势。

1 基本概念

本节主要对算法中使用的概念进行形式化定义, 如下所示。

定义 1 道路网络结构^[12,13]。给定城市路网数据, 可以抽象定义为一个有向图网络 $G=(V, E)$, $V=\{\Sigma v_i | i=1, 2, \dots, n\}$, $E=\{\Sigma e_i | i=1, 2, \dots, n\}$, 其中, V 代表顶点的集合, E 代表边的集合。

如图 1 所示, 在道路网络数据中, 一条完整的道路是由若干条路段构成, 路段由相邻两个折点连接而成。对路段折点进行存储, 通过搜索附近的路段折点得到相关的路段。

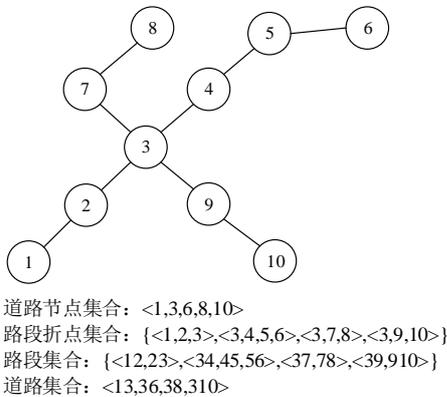


图 1 道路网络拓扑图

Fig. 1 Topological graph of road networks

定义 2 浮动车数据。给定一组浮动车数据 $T=(P_1, P_2, \dots, P_n)$, 其中每个点 P_i 是按照时间戳顺序生成的 GPS 轨迹点(待匹配点), 并且每个 P_i 包括车辆的 ID 编号、经纬度、时间、车辆角度、车辆速度等信息。多个待匹配点按时间顺序排列

而成的轨迹, 也称为 GPS 轨迹序列。

定义 3 候选路段。待匹配点 P_i 搜索附近道路, 得到候选路段集合。待匹配点 P_i 可能有一条或多条候选路段, 一个待匹配点的候选路段集合记为 $L_i=(l_1^i, l_2^i, \dots, l_m^i)$, $i=1,2,3,\dots,n$ 。

定义 4 GPS 候选点。待匹配点 P_i 与候选路段集合 L_i 中欧氏距离最近的点作为候选点, 待匹配点 P_i 的候选点集合记为 $C_i=(c_1^i, c_2^i, \dots, c_m^i)$, $i=1, 2, 3, \dots, n$ 。

2 地图匹配算法

2.1 工作原理

首先, 对原始道路数据进行预处理。合并所有类型的道路数据, 批量补充缺失字段值的道路数据, 建立拓扑规则对道路数据进行拓扑处理, 将处理后道路网络数据导入到数据库。其次, 对原始浮动车数据进行预处理。对冗余、异常和超出实验范围的浮动车数据进行剔除, 稀疏的 GPS 轨迹点使用插值补充的方法还原轨迹数据, 使用 Geohash 编码将二维的经纬度信息压缩为一维的位置表示, 将处理后 GPS 轨迹序列导入到数据库。下一步, 对待匹配点进行 Geohash 扩展搜索, 每个待匹配点得到多条候选路段。使用待匹配点的后向向量与路段向量形成的夹角进行筛选, 把待匹配点与筛选后的路段组成一个集合。最后, 对集合使用综合评价函数计算。当匹配次数低于 2 次时, 使用综合评价函数 1(如式(4)所示), 否则使用综合评价函数 2(如式(5)所示), 得到匹配结果。SP 曲线拟合地图匹配算法原理图如图 2 所示。

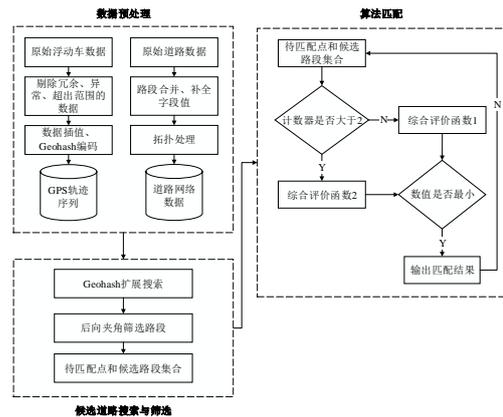


图 2 算法原理图

Fig. 2 Algorithm schematic

2.2 数据预处理

车辆在拥挤路段或者等待乘客的情况下, GPS 定位会出现重复的记录, 应对其进行删除处理。算法中对前一个 GPS 点和后一个 GPS 点求欧式距离, 距离小于 2 米对后一个 GPS 点进行剔除。一方面, 原始城市道路数据一般比较杂乱, 需要对道路数据进行路段合并以及拓扑处理, 缺失字段值的道路数据采用相邻道路数据求均值的方法进行补充, 使用 ArcMap 工具对城市道路进行处理得到道路网络数据。另一方面, 由于原始城市道路数据使用 GCJ-02 坐标系, GPS 轨迹序列数据使用 WGS-84 坐标系, 两者坐标系不一致, 需要对处理后的 GPS 轨迹序列从 WGS-84 转到 GCJ-02 坐标系。

当车辆进入 GPS 设备信号弱的区域, GPS 数据会产生漂移的情况产生异常记录, 需要对间隔较大的 GPS 序列进行插值补充全轨迹数据^[14]。使用车辆前一时刻的 GPS 瞬时速度和方向角进行插值, 插值原理如图 3 所示。

图 3 中黑色点为前一个时刻 GPS 定位点, 红色点为插值点, ε 为当前车辆的角度, 插值采用方法如式(1)所示。

$$\begin{aligned} x_{i+1} &= x_i + \Delta t \sin \varepsilon \\ y_{i+1} &= y_i + \Delta t \cos \varepsilon \end{aligned} \quad (1)$$

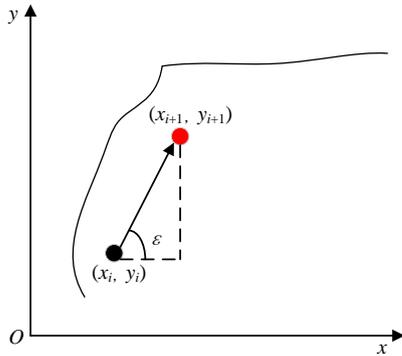


图 3 插值原理图

Fig. 3 Interpolation schematic

2.3 Geohash 编码及道路搜索

采用 Geohash 技术^[15,16]对路段折点和车辆待匹配轨迹序列进行编码, 把二维地理位置信息压缩成一维的位置信息进行存储并且对 Geohash 编码值进行索引构建^[17], 加快候选路段的搜索速度。

Geohash 是一个公共领域的地理编码系统, 可将经纬度编码成简短的字符串(东西、南北半球区间分别为[-180, 180]和[-90, 90])。其基本原理是将三维空间的地球理解成二维的平面, 使用二分法将平面递归分解成更小的区域, 每个区域范围内的经纬度都拥有相同的编码值。Geohash 编码的具体步骤如下:

- a) 分别对经度和纬度进行编码, 得到数字串, 经度编码方式如表 1 所示。
- b) 从第 0 位开始, 偶数位放经度码、奇数位放纬度码重组得到新的数字串。
- c) 以 5 位为一组, 使用 Base32 编码表(见表 2)对数字串转码得到 Geohash 编码值。

表 1 经度编码方式举例

Tab. 1 Example of longitude coding method

次数	经度范围	划分区间 0	划分区间 1	121.667
1	-180~180	-180~0	0~180	1
2	0~180	0~90	90~180	1
3	90~180	90~135	135~180	0
4	90~135	90~112.5	112.5~135	1
5	112.5~135	112.5~123.75	123.75~135	0
6	112.5~123.75	112.5~118.125	118.125~123.75	1

表 2 Base32 编码表

Tab. 2 The table of Base32 code

Decimal	0	1	2	3	4	5	6	7	8	9	10
Base32	0	1	2	3	4	5	6	7	8	9	b
Decimal	11	12	13	14	15	16	17	18	19	20	21
Base32	c	d	e	f	g	h	j	k	m	n	p
Decimal	22	23	24	25	26	27	28	29	30	31	
Base32	q	r	s	t	u	v	w	x	y	z	

例如, 图 4 中待匹配点的经纬度为(121.667, 31.1263), 7 级网格划分的经度编码为 110101101000010011, 纬度编码为 10101100010001001; 重新组码得到新的数字串 11100 11001 11100 01001 00000 01100 00111; 使用 Base32 转码得 wtw90d7。

本文道路搜索使用 Geohash 扩展的方法, 其原理是通过当前的 Geohash 区域反向计算得到附近 8 个区域, 区域内的道路为候选路段。道路搜索具体步骤如下:

- a) 使用当前 Geohash 编码值 Base32 反向计算得到数字串。
- b) 取偶数位为经度码、奇数位为纬度码, 分别进行经纬度码加减 1 操作。
- c) 对加减运算后的经纬度码重新组码。

- d) 使用 Base32 编码表转码, 得到附近 8 个区域的 Geohash 编码值。
- e) 查找数据库中和当前区域及附近 8 个区域的一样的 Geohash 编码值, 并把查询结果作为候选路段。

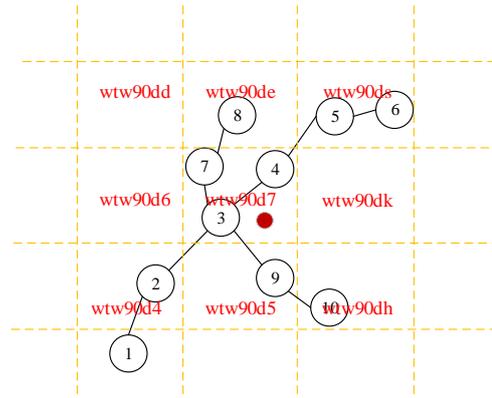


图 4 Geohash 位置图

Fig. 4 Figure of Geohash location

以图 4 中 wtw90d7 的左上角区域为例, 道路搜索步骤为: wtw90d7 反向计算得到数字串 11100 11001 11100 01001 00000 01100 00111; 取经度码并减 1 得 110101101000010010, 取纬度码并加 1 得 10101100010001010; 重新组码并 Base32 转码, 得到 wtw90dd 编码值; 数据库中查找与 wtw90d7 相同值, 把查询结果作为候选路段。

如图 4 所示, 棕红色点为待匹配点, 路段折点 3、4、7 和待匹配点位置在同一个区域 wtw90d7 中, 当前能搜索路段 3->7、3->4, 但不能搜索到 2->3、3->9 路段。因此需要对搜索范围增大, 以 wtw90d7 为中心, 扩展到附近 8 个搜索区域。以图 4 为例, 本次对待匹配点进行搜索的区域包括 wtw90dd、wtw90de、wtw90ds、wtw90d6、wtw90d7、wtw90dk、wtw90d4、wtw90d5、wtw90dh。

2.4 候选路段筛选

待匹配点对道路网络局部进行搜索后得到一条或多条候选道路, 为了进一步提高匹配效率, 需要对候选道路进行筛选^[18], 具体步骤如下:

- a) 对候选路段 l_i^m 与待匹配点求欧式距离, 如果距离大于 50 米, 剔除该候选道路。
- b) 当前待匹配点与下一时刻待匹配点构成向量 g , 沿着候选路段 l_i^m 车辆行驶方向构成向量 r , g 与 r 的后向夹角大于 30 度, 剔除该候选道路。候选路段分析原理如图 5 所示, 50 米范围内待匹配点 P_i 搜索到路段 1->2 和 2->3, 分别对应路段向量 r_1 和 r_2 。对后向向量 g 和路段向量 r 进行分析后, 剔除候选路段 2->3。

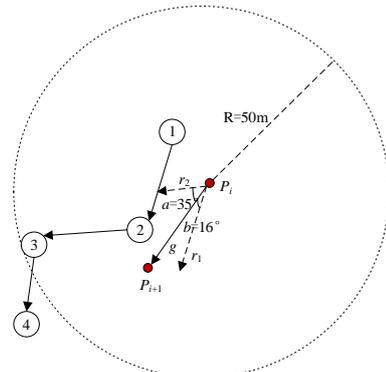


图 5 候选路段筛选示例

Fig. 5 Example of candidate segment selection

2.5 曲线拟合

实际应用中, 经常需要确定多个位置点之间的关系。在

已知变量 X 和 Y 的情况下, 可以对多个点构建一条平滑的曲线 $y=f(x)$, 即曲线拟合^[19]。在二维坐标系下, GPS 轨迹序列序列可以使用一个 m 次多项式 $y = w_0 + w_1x + w_2x^2 + \dots + w_mx^m$ 进行拟合, 得到一条平滑的曲线。为了求多项式的系数 w , 可以将多项式转换为矩阵的形式, 化简得到式(2)

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^m \\ 1 & x_2 & x_2^2 & \dots & x_2^m \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_n & x_n^2 & \dots & x_n^m \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \dots \\ w_m \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix} \quad (2)$$

由式(2)得到 $XA=Y$ 的形式, 可以进一步求得 $A=X^TY$, 即求得系数矩阵。

由于车辆使用低频 GPS 采样技术, 30s 左右获得一次定位记录, 点与点之间的距离相差较大, 采用 GPS 轨迹序列中 3 个连续的 GPS 点进行曲线拟合。拟合后曲线的斜率公式如下:

$$k = \frac{-\sum_{i=1}^3 x_i \sum_{j=1}^3 y_j + 3 \sum_{i=1}^3 x_i y_i}{3 \sum_{i=1}^3 x_i^2 - (\sum_{i=1}^3 x_i)^2} \quad (3)$$

2.6 综合评价函数

由于 GPS 设备存在定位误差, 车辆的 GPS 定位的位置不能完全准确地落在道路中, 例如在两条或两条以上的道路的位置, 车辆 GPS 定位点可能存在多条候选路段, 而距离定位点较近的道路不一定是正确的行驶道路。所以, 使用综合评价函数对候选匹配点进行计算, 得到正确的匹配结果。

待匹配点进行道路搜索后, 得到多个候选匹配点, 考虑车辆和道路速度因素^[20], 当已匹配点小于两个的时候, 由于不满足曲线拟合斜率的计算条件, 使用综合评价函数(如式(4)所示), 对候选点进行计算, 取代价最小的候选点作为匹配结果。

$$z = \alpha \frac{|k_{back} - k_r|}{Agl} + \beta \frac{d_{c^m \rightarrow P_i}}{R} + \gamma \left| \frac{Speed_c}{Speed_r} - 1 \right| \quad (4)$$

当已匹配点大于两个的时候, 使用综合评价函数 2(如式(5)所示), 对候选点进行计算。

$$z = \alpha \frac{\arctan(k_g) - \arctan(k_r)}{Agl} + \beta \frac{d_{c^m \rightarrow P_i}}{R} + \gamma \left| \frac{Speed_c}{Speed_r} - 1 \right| \quad (5)$$

其中, k_g 表示前 3 个 GPS 点的拟合曲线的斜率, k_r 是当前候选路段的斜率, k_{back} 是当前待匹配点的后向向量的斜率, $d_{c^m \rightarrow P_i}$ 是候选点 c^m 到待匹配点 P_i 的欧式距离, $Speed_c$ 表示车辆当前的瞬时速度, $Speed_r$ 是道路限制的速度, Agl 是 GPS 拟合曲线和候选道路夹角的阈值范围, 当夹角范围超过 30 度时, 匹配的错误率会增大, 故 Agl 设置为 30 度。 R 是道路搜索范围, R 设置为 50 米。对权重参数设置为 $\alpha=0.3, \beta=0.3, \gamma=0.4$ 。

3 基于曲线拟合的道路匹配算法

算法 1 基于曲线拟合的道路匹配算法

输入: GPS 轨迹序列 T , 道路网络数据 G 。

输出: 已匹配轨迹序列 H 。

1. $List = [], H = [];$
2. **for**(each p in T)
3. $RoadList = GeohashQuery(p, G);$
4. $RoadList = FiltrateRoad(RoadList);$
5. $List.add([p, RoadList]);$
6. **end for**
7. **for**(each l in $List$)
8. $p = l[0];$
9. $RoadList = l[1];$
10. $temList = [];$
11. **for**(each r in $RoadList$)
12. $temList.add(EvaluateFunction(p, r));$

13. **end for**
14. $H.add(\min(temList));$
15. **end for**
16. **return** $H;$

算法 1 的主要步骤:

a) 输入预处理后的 GPS 轨迹序列和道路网络数据, 创建 $List$ 集合临时记录待匹配点和候选路段, 创建 H 集合记录已匹配轨迹序列(第 1 行)。

b) 使用 $GeohashQuery()$ 函数对每个待匹配点 p 进行道路网络搜索扩展, 获取相应的候选道路集合 $RoadList$ (第 3 行)。获得的候选道路集合使用待匹配点 p 与候选路段后向夹角进行分析, 得到新的候选道路集合(第 4 行)。最后, 把待匹配点与新的道路网络集合封装成一个小的集合添加进 $List$ 集合里面(第 5 行)。

c) 取出 $List$ 集合的每一个小的集合, 进一步取出每个待匹配点 p 与相应的候选路段 $RoadList$ 集合(第 8~9 行)。由于一个待匹配点的候选路段可能有多条, 创建一个临时集合 $temList$ 记录评价函数的计算结果(第 10 行)。对候选路段 r 和待匹配点 p 使用评价函数 $EvaluateFunction()$ 得出数值结果(第 12 行)。一个待匹配点的候选路段使用综合评价函数计算后, 对多个计算结果使用 $\min()$ 函数, 获取数值最小的结果, 并且把待匹配点 p 匹配到数值最小的路段上(即新的经纬度), 得到匹配结果(第 12 行)。

d) 返回已匹配轨迹序列 H (第 16 行)。

算法改进后的优势在于: (a) 加快了候选道路的搜索, 主要体现在算法 1 中 $GeohashQuery$ 函数部分。与垂直投影算法和曲线拟合算法全局搜索的思想相比, 改进后的算法使用 $Geonhash$ 扩展搜索效率更高; (b) 降低了匹配的错误率和减少候选路段的计算量, 主要体现在算法 1 中 $FiltrateRoad$ 函数部分。一方面, $FiltrateRoad$ 函数过滤掉不相关的候选路段, 降低匹配错误的几率。另一方面, 过滤后的候选路段数量减少后, 计算量也随之减少, 进一步提升匹配的速度; (c) 提升了匹配的准确性, 对应算法 1 中 $EvaluateFunction$ 函数部分。评价函数综合考虑距离、轨迹角度和速度三者因素, 相比只考虑部分因素的垂直投影法和曲线拟合法, 改进搜的算法匹配准确率更高。

算法复杂度分析。GPS 轨迹序列包含 n 个待匹配点, 道路网络数据有 m 条路段, 对于每个待匹配点均需要对道路网络数据进行搜索。对 n 个待匹配点进行搜索, 则时间复杂度为 $O(n)$ 。当待匹配点搜索的路段数量为 0, 则时间复杂度为 $O(1)$, 这类搜索结果是没有意义的。当待匹配点搜索的路段数量大于 0 时, 则时间复杂度为 $O(\log m)$ 。所以, n 个待匹配点完成地图匹配的时间复杂度为 $O(n \log(m))$ 。

4 实验

4.1 实验数据及环境

4.1.1 浮动车数据

GPS 轨迹序列数据由两部分组成, 分别为 GPS 模拟数据和真实浮动车数据。GPS 模拟数据采用 GPS 误差原理, 随机产生服从高斯分布 20 米以内的经纬度误差, 共 1459 个待匹配 GPS 点。真实浮动车数据来源于香港科技大学提供的 2007 年 2 月 20 日上海市出租车 GPS 数据(<https://www.cse.ust.hk/scrg/>), 行走轨迹包含整个上海市以及上海周边城市, 数据采样频率大约 30s 至 60s, 数据量约 600 万条。从真实数据中选取 697 个记录作为待匹配 GPS 点, 利用人工查找的方法标注车辆行走的真实道路。

4.1.2 道路网路数据

实验中使用上海市道路网络数据, 道路类型包括国道、

省道、县道、乡道、城市快速路、高速和其他类型道路。为了验证提出的算法, 道路网络数据缩小至上海市 S20 外环高速以内的区域, 经过拓扑规则处理和拓扑构建后得到 32,916 条道路和 173,805 个路段折点, 采用高德地图 GCJ-02 坐标系, 经纬度跨度分别为: [121.347, 121.646], [31.123, 31.377]。

4.1.3 实验环境

使用 64 位 Windows 7 旗舰版的操作系统, Intel(R) Core(TM) i7-4770HQ CPU@2.20GHz 处理器和 16GB 的内存, 编程语言使用 Python3.7.2, 编程工具为 PyCharm2018.2.4, 数据库采用 PostgreSQL 10.7, 并通过 ArcMap10.4.1 工具对待匹配点、道路网络数据和匹配结果进行可视化。

4.2 评价指标

匹配算法的指标采用准确率(CMP^[20], Correct Matching Percentage)进行判定, 判定方法使用 ArcMap 工具的图层叠加分析^[21], 与真实道路相重合的匹配点则匹配正确, 该过程由图层叠加自动完成。式(6)中, Num_c 为正确匹配点的个数, Num_{all} 为待匹配点总数。

$$CMP = \frac{|Num_c|}{|Num_{all}|} \times 100\% \quad (6)$$

4.3 算法性能分析

为了验证所提出 SP 曲线拟合算法的性能优势, 实验中与传统的垂直投影算法和曲线拟合算法进行对比。数据进行预处理后对 GPS 模拟数据和真实浮动车数据进行插值补全, 插值补全的数据分别划分为 10s、30s、50s 和 70s 的采样频率, 分别在一条道路、两条道路和三条道路以上的道路数据进行验证。本文主要从道路准确性和时间性能分析两方面进行对比分析。

4.3.1 地图匹配准确性分析

采用随机抽样的方法, 分别对 GPS 模拟数据和真实浮动车数据进行实验, 采样频率相同的实验结果使用求均值的方法得到最终的准确匹配率 CMP。

图 6 表示模拟 GPS 数据分别在一条道路、两条道路和三

条道路以上使用不同算法的道路匹配准确率对比结果。实验结果表明: 在一条道路上进行实验, 三种算法的准确率较高, SP 曲线拟合算法表现最好, 准确率达到 94.79%以上。在两条道路上进行实验, 垂直投影和曲线拟合算法准确率开始下降, 而 SP 曲线拟合算法准确率仍然维持在 95.14%以上。在三条道路以上进行实验, SP 曲线拟合算法优势显著, 匹配率相对于垂直投影法平均提高了 27.41%, 相对于曲线拟合法提高了 30.24%。主要原因在于: 本文所提算法基于曲线拟合的思想, 筛选掉无关道路, 采用综合评价函数计算, 在一条、两条和三条道路以上的情况都能有较高的准确率。由于传统的垂直投影算法仅考虑空间欧式距离, 即便轨迹序列与理想的路线一致, 也会发生较高的错误率。曲线拟合算法由于仅考虑轨迹曲线和路段的角度, 虽然轨迹序列与理想路线相似, 所得的角度范围较小, 但是在两条和三条道路以上的情况, 平行的路段越多错误率也越高。

图 7 表示真实浮动车数据分别在一条道路、两条道路和三条道路以上使用不同算法的准确率对比实验结果。实验结果表明: 在一条道路上进行实验, SP 曲线拟合算法准确率维持在 89.38%以上, 垂直投影算法和曲线拟合算法准确率较低, 曲线拟合算法准确率最高只有 82.26%。在两条道路上进行实验, 曲线拟合算法表现稳定, 平均准确率只有 69.31%。垂直投影算法表现较差, 准确率保持在 61.25%~67.13%范围内。SP 曲线拟合算法表现最佳, 仍然具有较高的准确率, 准确率保持在 93.24%以上。在三条道路以上进行实验, 垂直投影算法和曲线拟合算法表现极差, 平均准确率分别只有 13.29%和 13.3%。SP 曲线拟合算法较好, 相比垂直投影算法提高了 70.04%, 相比曲线拟合法平均提高了 70.03%。原因在于: SP 曲线拟合算法使用后向夹角的思想过滤掉不相关路段, 且充分考虑车辆和道路的速度信息, 对路段的匹配结果更精确。相对于垂直投影算法利用空间距离思想和曲线拟合算法利用曲线斜率算法, 综合考虑多方面因素的 SP 曲线拟合算法有较高的准确率。

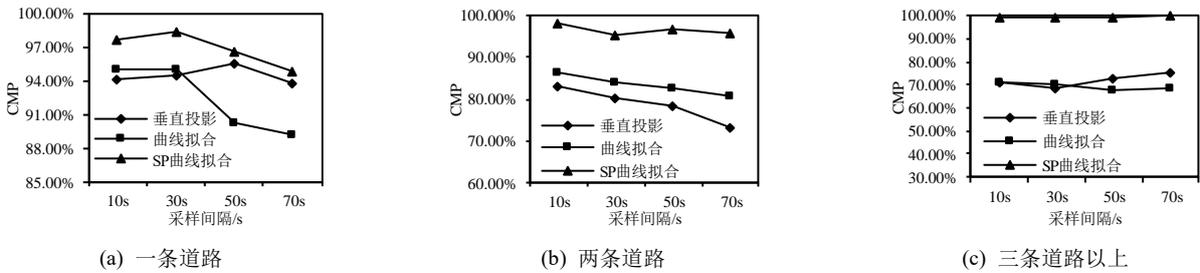


图 6 GPS 模拟数据准确率对比
Fig. 6 Compare of map-matching accuracy on GPS simulation data

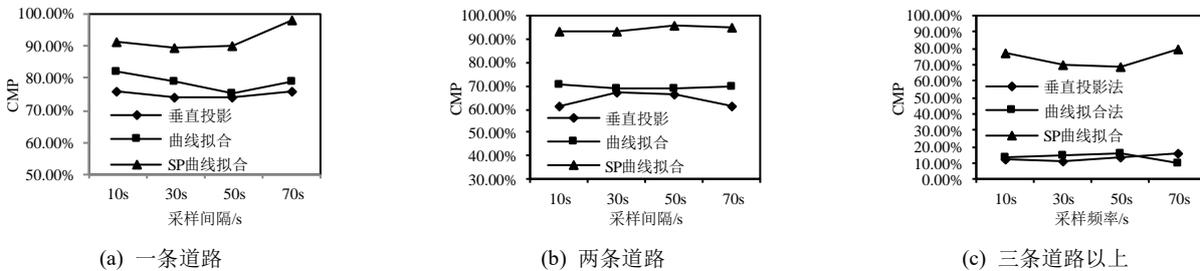


图 7 浮动车数据地图匹配准确率对比
Fig. 7 Compare of map-matching accuracy on floating car data

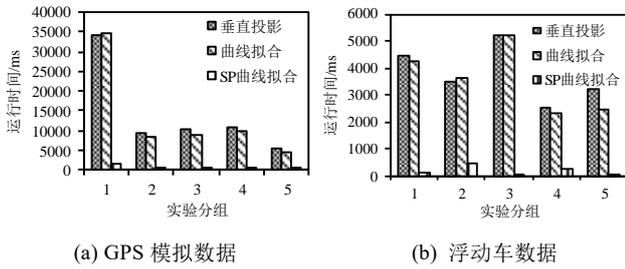
4.3.2 时间性能分析

采用随机抽样的方法, 分别在 GPS 模拟数据和真实浮动车数据中随机抽取 5 组数据对算法进行时间效率比较。随机抽取数据中, 一条道路、两条道路和三条道路以上都有一定的比例。

图 8 给出模拟 GPS 数据和真实浮动车数据使用不同算

法之间时间效率对比的实验结果, 实验结果表明: 垂直投影算法和曲线拟合算法时间效率相差不大, 平均单点匹配时间约为 0.9s, 时间复杂度均为 $O(nm)$ (n 表示待匹配 GPS 点个数, m 表示道路网络中路段的数量)。相对于垂直投影算法和曲线拟合算法, SP 曲线拟合算法运行时间显著下降, 平均单点匹配时间为 8ms, 时间复杂度为 $O(n \log(m))$ 。原因在于: 利用

Geohash 扩展搜索, 与全局遍历的思想不同, 待匹配点只搜索本区域和附近 8 个区域的路网折点, 搜索时间成本上从 $O(m)$ 降低至 $O(\log(m))$ 。此外, SP 曲线拟合算法采用 Geohash 技术搜索本区域内的路段和使用后向夹角对候选路段进行筛选, 时间效率得到大幅度提升。

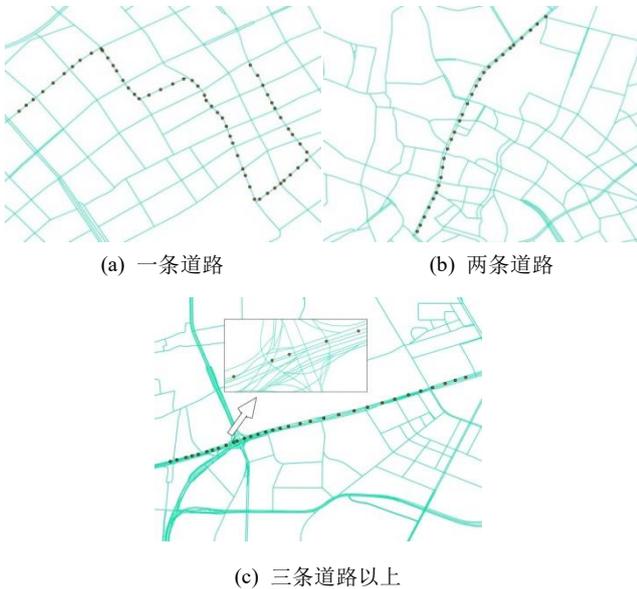


(a) GPS 模拟数据 (b) 浮动车数据
图 8 不同算法时间效率对比

Fig. 8 Time efficiency comparison of different algorithms

4.3.3 地图匹配效果

实验结果表明: SP 曲线拟合算法能准确地在一、两条和三条道路上进行地图匹配。实验中特别考虑在直线行驶、多次拐弯行驶和复杂环形道路行驶的情况, 实验效果证明 SP 曲线拟合算法具有很好的鲁棒性, 能适应以上不同的复杂情况。由图 9(a)可知, 车辆经过 5 次拐弯行驶, 轨迹点准确地被匹配到行驶路线上, 并没有被错误匹配到其他相邻的道路。由图 9(b)可知, 车辆行驶在单向 2 条车道的情况下(双向一共 4 条车道), 车辆行驶在正确的路线, 并没有出现在不同车道来回跳动的锯齿形现象。由图 9(c)可知, 在单向 4 车道情况下(环形交叉路口 8 车道以上), 车辆保持正常的行驶路线。SP 曲线拟合算法的精准性得益于综合考虑车辆与路段的夹角、距离和速度因素, 使用 Geohash 扩展搜索并剔除不相关道路, 高效准确地匹配地图。



(a) 一条道路 (b) 两条道路
(c) 三条道路以上
图 9 地图匹配效果

Fig. 9 Effect of Map-matching

5 结束语

为了提高地图匹配的正确率和效率, 针对低频 GPS 轨迹数据采样稀疏、不连贯的特点, 采用 Geohash 存储和搜索技术, 提出了一种改进曲线拟合的地图匹配方法, 充分考虑了车辆和道路的速度信息。在地图匹配算法基础上, 对稀疏的轨迹点进行插值, 还原了连续的车辆运动轨迹。通过实验的对比验证, 证明所提出地图匹配算法具有较高的正确率和匹配效率, 能满足当前地图匹配技术的要求。未来工作包括: a) 使用并行化的方法进行查询^[22]和插值, 在匹配阶段实现实

时地图匹配, 提高算法的效率; b) 对算法进行多方面测试, 提高算法的容错性和稳定性, 更好地应用于城市计算真实应用场景中。

参考文献:

- [1] Qiao Shaojie, Shen Dayong, Wang Xiaoteng, *et al.* A self-adaptive parameter selection trajectory prediction approach via hidden markov models [J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 16 (1): 284-296.
- [2] 高文超, 李国良, 塔娜. 路网匹配算法综述 [J]. 软件学报, 2018, 29 (02): 225-250. (Gao Wenchao, Li Guoliang, Ta Na. Survey of map matching algorithms [J]. Journal of Software, 2018, 29 (2): 225-250.)
- [3] Bernstein D, Kornhauser A. An introduction to map matching for personal navigation assistants [J]. Geometric Distributions, 1998, 122 (7): 1082-1083.
- [4] White C, Bernstein D, Kornhauser A. Some map matching algorithms for personal navigation assistants [J]. Transportation Research Part C: Emerging Technologies, 2000, 8 (1-6): 91-108.
- [5] Taylor G, Blewitt G, Steup D, *et al.* Road reduction filtering for GPS-GIS navigation [J]. Transactions in Gis, 2010, 5 (3): 193-207.
- [6] Alt H, Efrat A, Gyunter R, *et al.* Matching planar maps [J]. Journal of Algorithms, 2003, 49 (2): 262-283.
- [7] 盛彩英, 席唱白, 钱天陆, 等. 浮动车轨迹点地图匹配及插值算法 [J]. 测绘科学, 2019, 44 (08): 106-112. (Sheng Caiying, Xi Changbai, Qian Tianlu, *et al.* Study of map-matching and interpolation algorithm of floating car data [J]. Science of Surveying and Mapping, 2019, 44 (8): 106-112.)
- [8] Jabbour M, Bonnifait P, Cherfaoui V. Map-matching integrity using multi-sensor fusion and multi-hypothesis road tracking [J]. Journal of Molecular Biology, 2008, 9 (3): 598-600.
- [9] Nadine S, Kay W. Map matching of GPS traces on high-resolution navigation networks using the multiple hypothesis technique (MHT) [J]. Working Paper Transport and Spatial Planning, 2009, (10): 568-588.
- [10] Kim S, Kim J. Adaptive fuzzy-network-based C-measure map-matching algorithm for car navigation system [J]. IEEE Transactions on Industrial Electronics, 2001, 48 (2): 432-441.
- [11] Zhang L, Liu J, Wang R, *et al.* Trust evaluation model based on improved D-S evidence theory [J]. Journal on Communications, 2013, 34 (7): 167-173.
- [12] 邹方磊. 嵌入式 GIS 中多尺度道路网络路径规划方法的研究 [D]. 河南: 解放军信息工程大学, 2008. (Zou Fanglei. Research on the route planning method of multi-scale road networks in embedded GIS [D]. Henan: Information Engineering University, 2008.)
- [13] Qiao Shaojie, Han Nan, Gao Yunjun, *et al.* A fast parallel community discovery model on complex networks through approximate optimization [J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30 (9): 1638-1651.
- [14] 陈鹏. 广义延拓插值法在 GPS 数据处理中的应用研究 [D]. 湖南: 中南大学, 2009. (Chen Peng. The applied research of the generalized extension interpolation in GPS data processing [D]. Hunan: Central South University, 2009.)
- [15] 乔少杰, 韩楠, 李天瑞, 等. 一种基于空间编码技术的轨迹特征提取方法 [J]. 中国科学: 信息科学, 2017, 47 (11): 1523-1537. (Qiao Shaojie, Han Nan, Li Tianrui, *et al.* A trajectory feature extraction approach based on spatial coding technique [J]. SCIENTIA SINICA Informations, 2017, 47 (11): 1523-1537.)
- [16] Qiao Shaojie, Han Nan, Wang Junfeng, *et al.* Predicting long-term trajectories of connected vehicles via prefix-projection technique [J].

- IEEE Transactions on Intelligent Transportation Systems, 2017, 19 (7): 2305-2315.
- [17] Qiao Shaojie, Han Nan, Zhu W, *et al.* TraPlan: an effective three-in-one trajectory-prediction model in transportation networks [J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 16 (3): 1188-1198.
- [18] 嘉邨, 孙立双, 王晓明. 北京出租车 GPS 轨迹数据地图匹配算法研究 [J]. 北京测绘, 2019, 33 (03): 255-260. (Jia Li, Sun Lishuang, Wang Xiaoming. Research on GPS track data map matching algorithm of Beijing [J]. Beijing Surveying and Mapping, 2019, 33 (3): 255-260.)
- [19] 周颖, 程荫杭. 基于曲线拟合的地图匹配算法 [J]. 交通运输系统工程与信息, 2004 (02): 68-70. (Zhou Ying, Cheng Yinhang. Map-matching algorithm based on curve-fitting model [J]. Journal of Transportation Systems Engineering and Information Technology, 2019, 33 (3): 255-260.)
- [20] Zhang Yaying, He Yulong. An advanced interactive-voting based map matching algorithm for low-sampling-rate GPS data [C]// IEEE 15th International Conference on Networking, Sensing and Control (ICNSC), China: Zhuhai, 2018: 1-7.
- [21] 陈艳华, 张琳. 基于 GIS 的图层叠加法在地震次生火灾区划单元中的应用 [J]. 河北联合大学学报: 自然科学版, 2014, 36 (02): 124-128. (Chen Yanhua, Zhang Lin. Application of overlapping layer method on regionalizing units of post-earthquake fire based on GIS [J]. Journal of Hebei Polytechnic University: Social Science Edition, 2014, 36 (2): 124-128.)
- [22] 罗徐玲. 基于知识库与云平台的海量数据存储与查询设计与实现 [D]. 北京: 北京邮电大学, 2017. (Luo Xuling. The design application of storage and query of big data based on cloud platform [D]. Beijing: Beijing University of Posts and Telecommunications, 2017.)